Introduction to Machine Learning

NYU K12 STEM Education: Machine Learning

Department of Electrical and Computer Engineering,
NYU Tandon School of Engineering
Brooklyn, New York

Course Details

- Course Website
- Instructors:

Rugved Mhatre
rugved.mhatre@nyu.edu

Akshath Mahajan
akshathmahajan@nyu.edu

NYU

Outline

Outline

The XOR Problem

- What is XOR?

The XOR Problem

- What is XOR?
  The logical operation eXclusive-OR outptus 1 when inputs differ, and 0 otherwise.

Input A	Input B	Output
0	0	0
0	1	1
1	0	1
1	1	0

Table 1: XOR Truth Table

The XOR Problem

- What is XOR?
  The logical operation eXclusive-OR outptus 1 when inputs differ, and 0 otherwise.

Input A	Input B	Output
0	0	0
0	1	1
1	0	1
1	1	0

Table 1: XOR Truth Table

- Why is this a problem?

The XOR Problem

- Let’s see the decision boundary for AND and OR gates graphically

The XOR Problem

- Let's see the decision boundary for AND and OR gates graphically

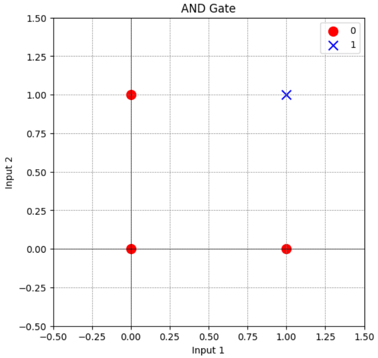


Figure 1: AND Gate

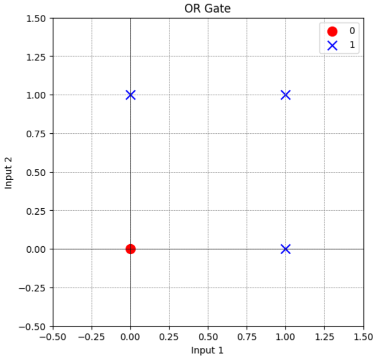


Figure 2: OR Gate

The XOR Problem

- Let's see the decision boundary for AND and OR gates graphically

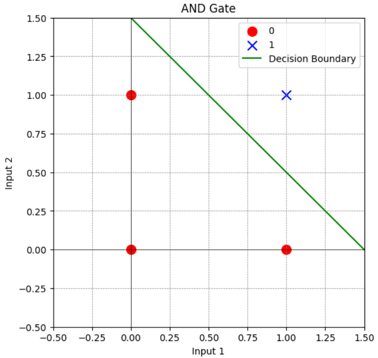


Figure 1: AND Gate

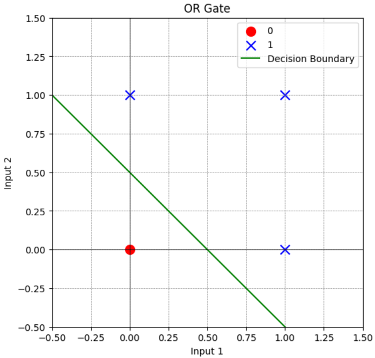


Figure 2: OR Gate

The XOR Problem

- Let's see the decision boundary for AND and OR gates graphically

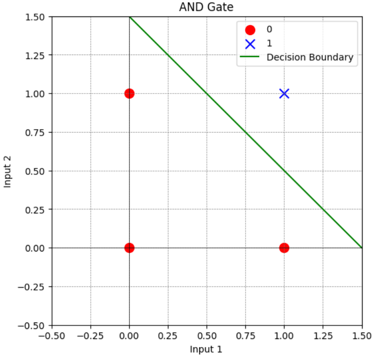


Figure 1: AND Gate

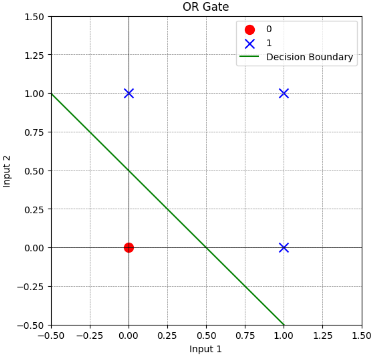


Figure 2: OR Gate

- What about the XOR gate?

The XOR Problem

- What about the XOR gate?

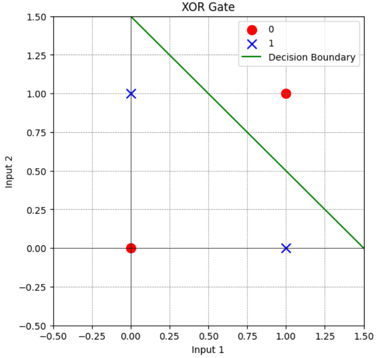


Figure 3: XOR Gate

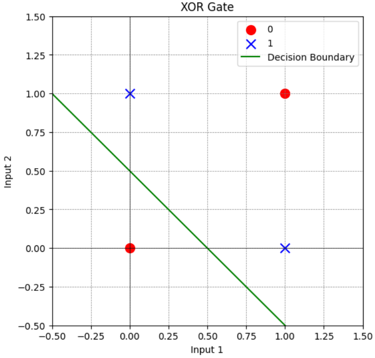


Figure 4: XOR Gate

Describing the General Limitation

- What about other distribution shapes?

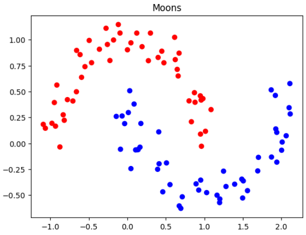


Figure 5: Moons

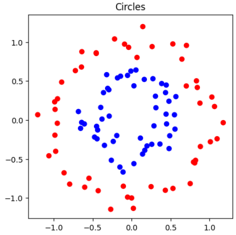


Figure 6: Cirles

Describing the General Limitation

- What about other distribution shapes?

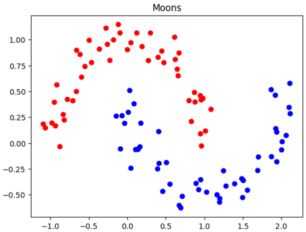


Figure 5: Moons

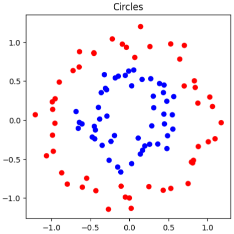


Figure 6: Cirles

- Can you suggest other shapes?

Describing the General Limitation

- What about other distribution shapes?

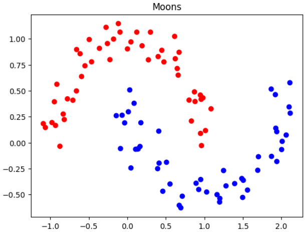


Figure 5: Moons

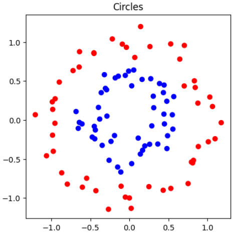


Figure 6: Cirles

- Can you suggest other shapes?
- What can we do about this?

Describing the General Limitation

- What about other distribution shapes?

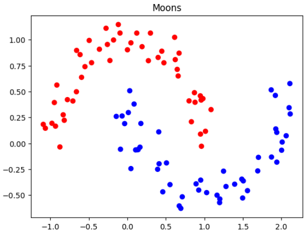


Figure 5: Moons

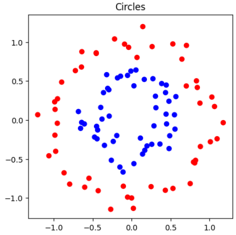


Figure 6: Cirles

- Can you suggest other shapes?
- What can we do about this?
  - Non-Linear classifiers?

Describing the General Limitation

- What about other distribution shapes?

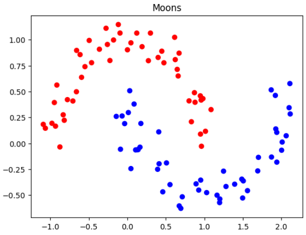


Figure 5: Moons

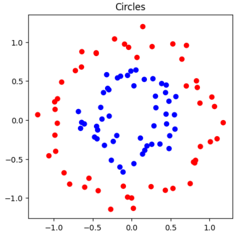


Figure 6: Cirles

- Can you suggest other shapes?
- What can we do about this?
  - Non-Linear classifiers?
  - Enter Neural Networks

Outline

1. Review
2. Limitations of Linear Classifiers
3. Neural Networks
4. Stochastic Gradient Descent
5. Overparameterized Models

Neurons

- What is a Neuron?

Neurons

- What is a Neuron?
- There are 2 definitions

Neurons

- What is a Neuron?
- There are 2 definitions
  - Biological Neuron

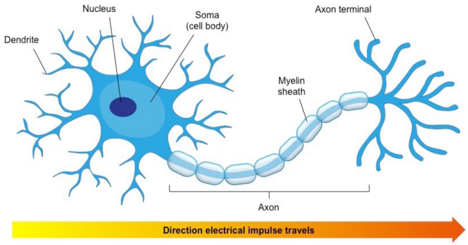


Figure 7: Biological Neuron
Source: Arizona State University

Neurons

- What is a Neuron?
- There are 2 definitions
  - Biological Neuron
  - Mathematical Neuron (Perceptron)

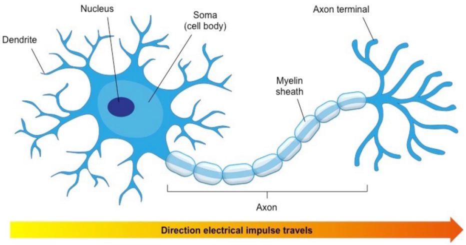


Figure 7: Biological Neuron
Source: Arizona State University

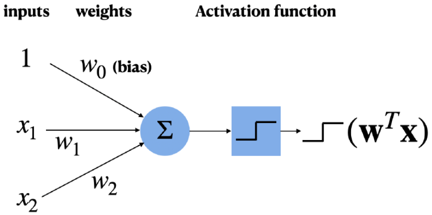
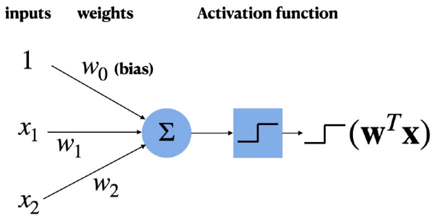
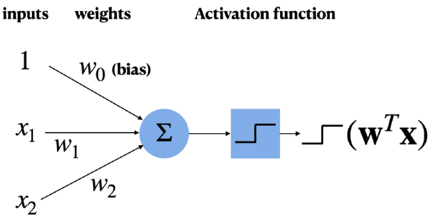


Figure 8: Mathematical Neuron

The Perceptron



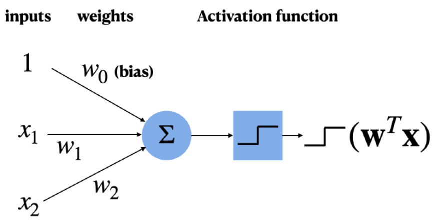
The Perceptron



$$y = \varphi(\mathbf{W}^T \cdot \mathbf{X})$$

- Looks similar to Linear Classification!

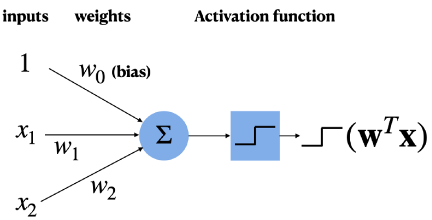
The Perceptron



$$y = \varphi(\mathbf{W}^T \cdot \mathbf{X})$$

- Looks similar to Linear Classification!
- How is this supposed to revolutionize Machine Learning?

The Perceptron



$$y = \varphi(\mathbf{W}^T \cdot \mathbf{X})$$

- Looks similar to Linear Classification!
- How is this supposed to revolutionize Machine Learning?
- HINT: How many neurons are in your brain?
  Does the Activation need to be Logistic/Sigmoid?

Neural Networks

- Solution 1: Connect many neurons together!
- This is the basic concept of a neural network
- Let's see a Multi-Layer Perceptron/Fully Connected Feed-Forward Network

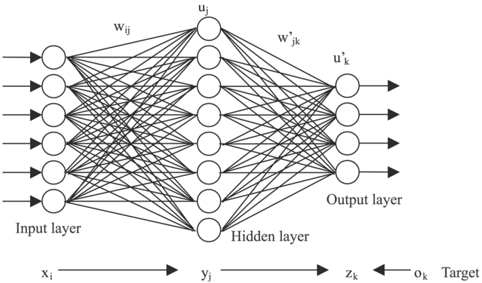


Figure 9: Neural Network

Activation Functions

- Solution 2: Use different Activation Functions

Activation Functions

- Solution 2: Use different Activation Functions
- These have a significant impact on the behavior of a Neuron

Activation Functions

- Solution 2: Use different Activation Functions
- These have a significant impact on the behavior of a Neuron

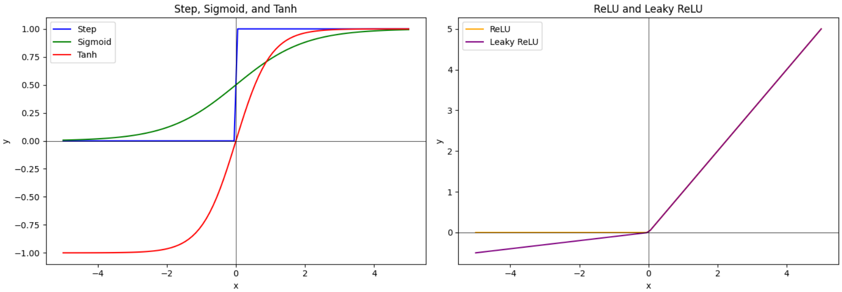


Figure 10: Different Activation Functions

Activation Functions

- Solution 2: Use different Activation Functions
- These have a significant impact on the behavior of a Neuron
- Softmax activation is particularly important!

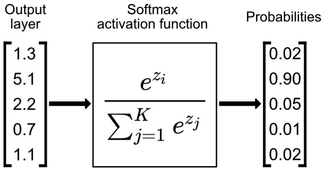


Figure 10: Softmax Activation
Source: Towards Data Science

MLP Example - 1

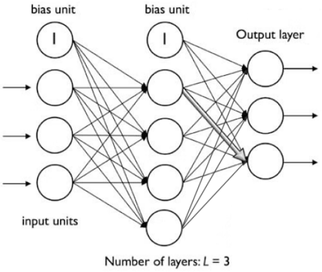


Figure 11: MLP Example 1

- What is the shape of input and output?

MLP Example - 1

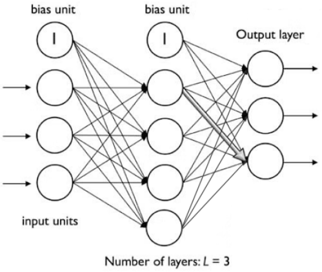


Figure 11: MLP Example 1

- What is the shape of input and output?

MLP Example - 1

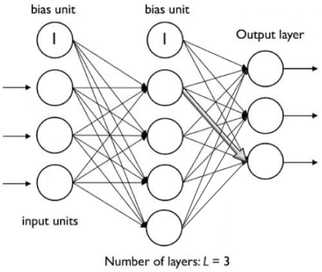


Figure 11: MLP Example 1

- What is the shape of input and output? (3, 1) and (3, 1)
- How many parameters does the model have?

MLP Example - 1

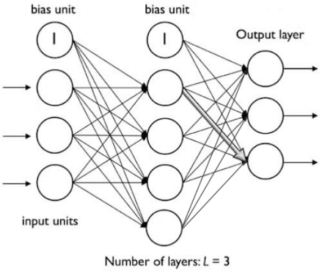


Figure 11: MLP Example 1

- What is the shape of input and output? (3, 1) and (3, 1)
- How many parameters does the model have? 31
- What activation functions would you use for output layer?

MLP Example - 1

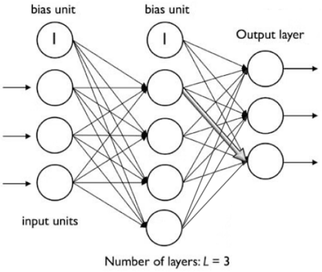


Figure 11: MLP Example 1

- What is the shape of input and output? (3, 1) and (3, 1)
- How many parameters does the model have? 31
- What activation functions would you use for output layer? Softmax

MLP Example - 2

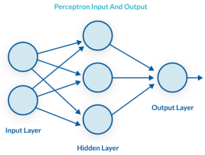


Figure 12: MLP Example 2

- What is the shape of input and output?
- How many parameters does the model have?
- What activation functions would you use for output layer?

MLP Example - 2

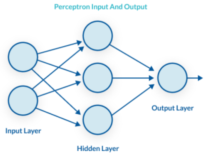


Figure 12: MLP Example 2

- What is the shape of input and output? (2, 1) and (1, 1)
- How many parameters does the model have? 13
- What activation functions would you use for output layer? Depends on the task

Deep Neural Networks

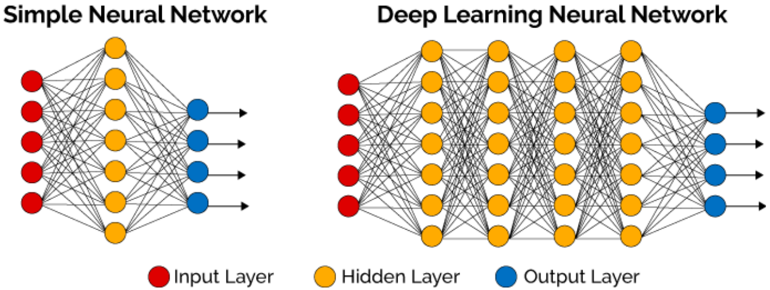


Figure 13: Simple vs Deep Networks

- There are many choices for the number of hidden layers and number of neurons per layer

Deep Neural Networks

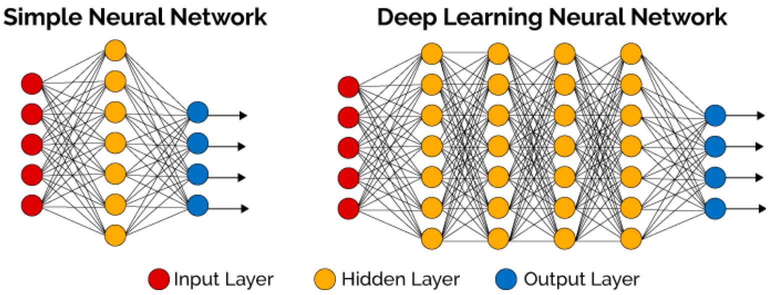


Figure 13: Simple vs Deep Networks

- There are many choices for the number of hidden layers and number of neurons per layer
- MLPs can approximate almost any continuous function

Deep Learning

- What does deep learning mean?
  - Deep: Neural network architectures with many hidden layers
  - Learning: Optimizing model parameters given a dataset

Deep Learning

- What does deep learning mean?
  - Deep: Neural network architectures with many hidden layers
  - Learning: Optimizing model parameters given a dataset
- Generally, deeper models have more parameters and require larger datasets to learn

Deep Learning

- What does deep learning mean?
  - Deep: Neural network architectures with many hidden layers
  - Learning: Optimizing model parameters given a dataset
- Generally, deeper models have more parameters and require larger datasets to learn
- What problems can we expect?

Deep Learning

- What does deep learning mean?
  - Deep: Neural network architectures with many hidden layers
  - Learning: Optimizing model parameters given a dataset
- Generally, deeper models have more parameters and require larger datasets to learn
- What problems can we expect?
  - Overfitting

Deep Learning

- What does deep learning mean?
  - Deep: Neural network architectures with many hidden layers
  - Learning: Optimizing model parameters given a dataset
- Generally, deeper models have more parameters and require larger datasets to learn
- What problems can we expect?
  - Overfitting
  - Computational Limitations

Outline

Computational Limitations

- For deep learning systems to perform well, we need large datasets -
  - COCO - 330K images (25 GB)
  - ImageNet - 14 million images (300 GB)

Computational Limitations

- For deep learning systems to perform well, we need large datasets -
  - COCO - 330K images (25 GB)
  - ImageNet - 14 million images (300 GB)
- Computational Challenges

Computational Limitations

- For deep learning systems to perform well, we need large datasets -
  - COCO - 330K images (25 GB)
  - ImageNet - 14 million images (300 GB)
- Computational Challenges
  - Memory Limitation - GeForce RTX 2080 Ti has 11 GB memory, while ImageNet is about 300 GB.

Computational Limitations

- For deep learning systems to perform well, we need large datasets -
  - COCO - 330K images (25 GB)
  - ImageNet - 14 million images (300 GB)
- Computational Challenges
  - Memory Limitation - GeForce RTX 2080 Ti has 11 GB memory, while ImageNet is about 300 GB.
  - Computation - Calculating gradients for the whole dataset is slow and done several times.

Stochastic Gradient Descent

- We don't really need to calculate gradients from the whole data

Stochastic Gradient Descent

- We don’t really need to calculate gradients from the whole data
- Calculate gradients from subsets of the whole dataset, one at a time

Stochastic Gradient Descent

- We don't really need to calculate gradients from the whole data
- Calculate gradients from subsets of the whole dataset, one at a time
  - The subset can fit in memory
  - The gradient of subset is calculated fast

Stochastic Gradient Descent

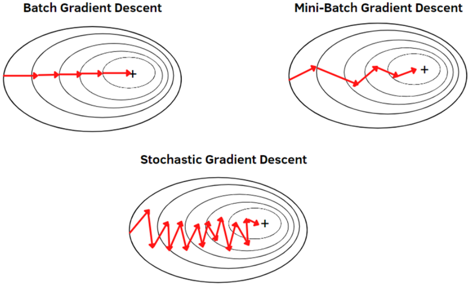
- We don't really need to calculate gradients from the whole data
- Calculate gradients from subsets of the whole dataset, one at a time
  - The subset can fit in memory
  - The gradient of subset is calculated fast
- But there is a tradeoff:

Stochastic Gradient Descent

- We don't really need to calculate gradients from the whole data
- Calculate gradients from subsets of the whole dataset, one at a time
  - The subset can fit in memory
  - The gradient of subset is calculated fast
- But there is a tradeoff:
  - Each gradient is a bit noisy
  - More number of gradients need to be calculated

Stochastic Gradient Descent

- The descent ends up looking like this -



Stochastic Gradient Descent

- Consider a subset of the original dataset having size B
- The loss is then calculated as -

$$L(W) = \frac{1}{B}\sum_{i=1}^{B}(y_i - \hat{y}_i)^2$$

- The weight update rule then becomes -

$$W_{new} = W - \alpha \nabla L(W)$$

Stochastic Gradient Descent

- For different sizes of B, we have -
  - SGD: $B = 1$, and results in very noisy gradients
  - Mini-batch GD: B is small (typically 32, 64, 128 for images), and gradients have some noise
  - GD: $B = N$, and gradients have no noise

Stochastic Gradient Descent

- For different sizes of B, we have -
  - SGD: $B = 1$, and results in very noisy gradients
  - Mini-batch GD: B is small (typically 32, 64, 128 for images), and gradients have some noise
  - GD: $B = N$, and gradients have no noise
- Even if feasible, GD is not a good idea. Noisy gradients can help
  - escape from local minima
  - escape from saddle points
  - improve generalization

Outline

Overparameterized Models

- Modern deep learning models are heavily overparameterized

Overparameterized Models

- Modern deep learning models are heavily overparameterized
  - GPT-3: State-of-the-art language model, 175 billion parameters
  - ResNet: State-of-the-art vision model, 10-60 million parameters

Overparameterized Models

- Modern deep learning models are heavily overparameterized
  - GPT-3: State-of-the-art language model, 175 billion parameters
  - ResNet: State-of-the-art vision model, 10-60 million parameters
- Conventional wisdom: Such models overfit.

Overparameterized Models

- Modern deep learning models are heavily overparameterized
  - GPT-3: State-of-the-art language model, 175 billion parameters
  - ResNet: State-of-the-art vision model, 10-60 million parameters
- Conventional wisdom: Such models overfit.
- It is not the case in practice!